7К

УДК 004.81:159.953.52

Я.С. Коровин, С.Н. Матвеев
НИИ многопроцессорных вычислительных систем имени академика А.В. Каляева
Южного федерального университета, г. Таганрог, Россия
НГДУ «Комсомольскнефть», г. Сургут, Россия
korovin@mvs.tsure.ru, matveev_sn@ngdukn.surgutneftegas.ru

Методика объяснения нейросетевого вывода. О подходе к решению проблемы дефицита обучающих примеров

Предложена и исследована методика объяснения нейросетевых решений. Рассмотрен подход к решению проблемы дефицита обучающих примеров на примере предсказания роста биржевых котировок.

Введение

Целью данной статьи является решение двух актуальных проблем нейроинформатики: объяснения решений нейронной сети и решения проблемы недостатка обучающих примеров при операции нейросетевого предсказания.

Нейронные сети (НС) в отличие от экспертных систем (ЭС) позволяют одновременно анализировать множество параметров и не требуют при этом явной формализации правил вывода. В то же время технология нейронных сетей не представляет возможным проследить всю цепочку вывода (т.к. нейросетевое распознавание является аналогом операции сравнения с эталоном), но и логику принятия итогового решения как такового в итоге. В связи с этим объяснение решений нейронной сети является актуальной научной проблемой нейроинформатики. В статье предлагается подход к ее решению.

Наряду с этим в вопросе нейросетевого предсказания нередко «слабым звеном» выступает фактор дефицита примеров для обучения нейронной сети. Рассматривается и исследуется методика генерации искусственных примеров, получаемых из уже имеющихся применением к ним различного рода преобразований на примере предсказания биржевых котировок.

1. Извлечение правил из нейронных сетей

Пусть A обозначает набор из N свойств A1,A2...AN, а {a} – множество возможных значений, которое может принимать свойство Ai. Обозначим через C множество классов c1,c2...cN. Для обучающей выборки известны ассоциированные пары векторов входных и выходных значений (a1...am,ck), где $ck \in C$. Алгоритм извлечения разделяющих правил включает три этапа:

1. Обучение нейронной сети. На этом данном этапе многослойный перцептрон обучается до получения требуемого уровня качества распознавания.

2. Прореживание нейронной сети. Обученная нейронная сеть содержит все возможные связи между входными нейронами и нейронами скрытого слоя, а также между последними и выходными нейронами. Полное число этих связей обычно столь велико, что из анализа их значений невозможно извлечь обозримые для пользователя классифицирующие правила. Прореживание заключается в удалении излишних связей и нейронов,

не приводящем к увеличению ошибки классификации сетью. Результирующая сеть обычно содержит немного нейронов и связей между ними и ее функционирование поддается исследованию.

3. Извлечение правил. На этом этапе из прореженной нейронной сети извлекаются правила, имеющие форму «если (a1 ⊗ q1) и (a2 ⊗ q2) и ... и (an ⊗ qn), то», где – константы, ⊗ – оператор отношения (=, ≥, ≤, <). Предполагается, что эти правила достаточно очевидны при проверке и легко применяются к БД больших размерностей.

1.1. Обучение нейронной сети

Предположим, что обучающий набор данных Z необходимо разбить на два класса A и B. В этом случае сеть должна содержать N входных и 2 выходных нейрона. Каждому из классов будут соответствовать следующие активности выходных нейронов (1,0) и (0,1). В качестве функции активации промежуточных нейронов используется гиперболический тангенс, так что их состояния изменяются в интервале [-1,1]. В то же время функцией активации выходных нейронов является функция Ферми (состояния в интервале [0,1]). Обозначим через $o_i^k, (i = 1,2)$ состояния выходных нейронов при предъявлении на вход сети вектора признаков *k*-го объекта x^k. Будем считать, что этот объект правильно классифицирован сетью, если

$$\max_i \left| o_i^k - t_i^k \right| \le \eta_1 ,$$

где: $t_1^k = 1$, если $x^k \in A$ и $t_2^k = 1$ если $x^k \in B$, а $0 < \eta < 0,5$. В остальных случаях $t_1^k = 0$.

Минимизируемая функция ошибки должна не только направлять процесс обучения в сторону правильной классификации всех объектов обучающей выборки, но и делать малыми значения многих связей в сети, чтобы облегчить процесс их прореживания.

$$E = E_0 + \varepsilon E_1 ,$$

где

$$E_0 = -\sum_k \sum_i (t_i^k \log o_i^k + (1 - o_i^k)\log(1 - o_i^k))$$

функция взаимной энтропии, минимизация которой происходит быстрее, чем минимизация среднеквадратичной ошибки. Штрафная функция:

$$E_1 = \sum_{j=1}^{N} \sum_{l=1}^{N} A \frac{(w_{lj}^h)^2}{1 + (w_{lj}^h)^2} + \sum_{l=1}^{N_h} \sum_{i=1}^{2} \frac{(w_{il}^o)^2}{1 + (w_{il}^o)^2} .$$

Здесь N_h – число нейронов в скрытом слое, w_{li}^h – величина связи между j-м входным и l-м скрытым нейронами, w_{il}^o – вес связи между l-м скрытым и i-м выходным нейронами [1].

Использование регуляризирующего члена E1 приводит к дифференциации весов по величинам, уменьшая большинство, но сохраняя значения некоторых из них. Обучение сети производится методом обратного распространения ошибки.

1.2. Прореживание нейронной сети

Полное число связей в обученной сети составляет $(N + N_o)N_h$. Можно показать, что связь между входным и промежуточным нейроном w_{lj}^h можно удалить без снижения точности классификации сетью при выполнении условий $\max_i \left| w_{i,l}^o w_{l,j}^h \right| \le \pm\eta_2$ и $\eta_1 + \eta_2 < 0,5$. Аналогичным образом, удаление связи $w_{i,l}^o$ не влияет на качество классификации, если $w_{i,l}^o \le \pm\eta_2$.

1.3. Извлечение правил

1. Выбирается значение параметра $\varepsilon \in (0,1)$, управляющего числом кластеров активности нейрона скрытого слоя. Пусть h_1 – активность этого нейрона при предъявлении сети первого вектора обучающего набора. Положим число кластеров $N_{clust} = 1$, положение кластера $A_{clust}(1)h_1, count(1) = 1, sum(1) = h_1$.

2. Для всех векторов выборки обучающих примеров $k = 1,...K$
– определяется активность нейрона скрытого слоя h,
– если существует индекс j, такой что

$$\left|h - A_{clust}(j)\right| = \min_{j \in \{1,...,N_{clust}\}} \left|h - A_{clust}(j)\right| \text{ и}$$

$$\left|h - A_{clust}(j)\right| \le \varepsilon,$$

то

$$count(j) := count(j) + 1, sum(N_{clust}) := sum(N_{clust}) + h,$$

иначе

$$N_{clust} = N_{clust} + 1, A_{clust}(N_{clust}) = h,$$
$$count(N_{clust}) = 1, sum(N_{clust}) = h.$$

3. Заменить A_{clust} на среднее значение активаций нейрона, объединенных в один и тот же кластер:

$$A_{clust}(j) = sum(j) / count(j), j_1, ..., N_{clust}.$$

4. Проверить точность классификации объектов сетью при замене истинных значений активации нейрона скрытого слоя на $A_{clust}(j)$.

5. Если точность классификации оказалась ниже заданного значения, то уменьшить значение ε и вернуться к шагу 1.

2. Методика решения проблемы дефицита обучающих примеров

Рассмотрим предлагаемую методику на конкретном примере нейросетевого предсказания биржевых котировок.

Предположим, что имеется кривая роста котировок за предыдущий период (банковский день, сессию и т.д.). Соответствие данных по осям выступает как выборка обучающих примеров, однако их количество недостаточно для качественного обучения нейронной сети. Используя опыт биржевых экспертов, можно заключить, что в основном игроки обращают внимание на форму кривой цен, а не на конкретные значения по осям. Поэтому если немного «растянуть» по оси котировок весь временной ряд, то полученный в результате такого преобразования ряд также можно использовать для обучения наряду с исходным. Таким образом, увеличивается число примеров за счет использования априорной информации, вытекающей из психологических особенностей восприятия временных рядов участниками рынка.

Еще один способ решения упомянутой выше проблемы в области предсказания состояния рынка – это так называемое использование скрытой симметрии в валютной торговле. Смысл этой симметрии в том, что валютные котировки могут рассматриваться с двух точек зрения, например как ряд DM/$ или как ряд $/DM. Возрастание одного из них

соответствует уменьшению другого. Это свойство можно использовать для удвоения числа примеров: каждому примеру вида $(X_{t-d+1},...,X_{t-1},X_t) \to X_{t+1}$ можно добавить его симметричный аналог $(-X_{t-d+1},...,-X_{t-1},-X_t) \to -X_{t+1}$. Эксперименты по нейросетевому предсказанию показали, что для основных валютных рынков учет симметрии поднимает норму прибыли примерно в два раза, конкретно – с 5 % годовых до 10 % годовых, с учетом реальных транзакционных издержек [2].

2.1. Измерение качества предсказаний

Хотя предсказание финансовых рядов и сводится к задаче аппроксимации многомерной функции, оно имеет свои особенности, как при формировании входов, так и при выборе выходов нейросети. Первый аспект, касающийся входов, мы уже обсудили. Теперь коснемся особенностей выбора выходных переменных. Но прежде ответим на главный вопрос: как измерить качество финансовых предсказаний. Это поможет определить наилучшую стратегию обучения нейросети.

2.2. Связь предсказуемости с нормой прибыли

Особенностью предсказания финансовых временных рядов является стремление к получению максимальной прибыли, а не минимизации среднеквадратичного отклонения, как это принято в случае аппроксимации функций.

В простейшем случае ежедневной торговли прибыль зависит от верно угаданного знака изменения котировки. Поэтому нейросеть нужно ориентировать именно на точность угадывания знака, а не самого значения. Найдем, как связана норма прибыли с точностью определения знака в простейшей постановке ежедневного вхождения в рынок [2].

Обозначим на момент t: полный капитал игрока K_t, относительное изменение котировки $x_t = \Delta C_t / C_t$, а в качестве выхода сети возьмем степень ее уверенности в знаке этого изменения $y_t \in [-1,1]$. Такая сеть с выходной нелинейностью вида $y = tg(\alpha)$ обучается предсказывать знак изменения и выдает прогноз знака с амплитудой, пропорциональной его вероятности. Тогда возрастание капитала на шаге t примет вид:

$$K_t = K_{t-1}[1 + |x_t|(x_t, y_t)],$$

где δ – доля капитала, «в игре». Выигрыш за все время игры:

$$K_t = K_0 \exp(\sum_{k=1}^{t} \ln[1 + x_k(y_k)])$$

нам и предстоит максимизировать, выбрав оптимальный размер ставок σ. Пусть в среднем игрок угадывает долю $p = \frac{1}{2} + \varepsilon$ знаков и соответственно ошибается с вероятностью $q = \frac{1}{2} - \varepsilon$. Тогда логарифм нормы прибыли:

$$\langle \ln(K_t / K_0) \rangle = t \langle p \ln(1 + |x|\delta) + q \ln(1 - |x|\delta \rangle,$$

а следовательно и сама прибыль, будет максимальным при значении

$$\delta = (p - q)\langle |x| \rangle^2$$

и составит в среднем:

$$\langle \ln(K_t / K_0) \rangle \approx t(p-q)^2 \frac{\langle |x| \rangle^2}{2\langle x^2 \rangle} = 2\alpha t \varepsilon^2 .$$

Здесь мы ввели коэффициент $\alpha = \langle |x| \rangle^2 / \langle x \rangle^2 \le 1$. Например, для Гауссова распределения $\alpha \approx 0{,}8$.

В итоге получаем следующую оценку нормы прибыли при заданной величине предсказуемости знака *I*, выраженной в битах:

$$K_t = K_0 2^{\alpha I t} .$$

То есть для ряда с предсказуемостью *I* в принципе возможно удвоить капитал за $t = 1/(\alpha I)$ вхождений в рынок. Таким образом, даже небольшая предсказуемость знака изменения котировок способна обеспечить весьма заметную норму прибыли.

Подчеркнем, что оптимальная норма прибыли требует достаточно аккуратной игры, когда при каждом вхождении в рынок игрок рискует строго определенной долей капитала:

$$\langle \Delta K \rangle / K = \delta \langle |x| \rangle = (p-q) \langle |x| \rangle^2 / \langle x^2 \rangle = 2\alpha\varepsilon \approx 1.6\varepsilon,$$

где ΔK – типичная при данной ситуации рынка $\langle |x| \rangle$ величина выигрыша или проигрыша [2]. Как меньшие, так и большие значения ставок уменьшают прибыль. Причем чересчур рискованная игра может привести к проигрышу при любой предсказательной способности [3].

2.3. Выбор функционала ошибки

Если принять, что целью предсказаний финансовых временных рядов является максимизация прибыли, логично настраивать нейросеть именно на этот конечный результат. Например, при игре по описанной выше схеме для обучения нейросети можно выбрать следующую функцию ошибки обучения, усредненную по всем примерам из обучающей выборки:

$$E = -\langle \ln[1 + x_t \delta_t \,\mathrm{sgn}(y_t)] \rangle .$$

Здесь доля капитала в игре введена в качестве дополнительного выхода сети, настраиваемого в процессе обучения. При таком подходе первый нейрон, y_t, с функцией активации $f = tg(\delta)$ даст вероятность возрастания или убывания курса, в то время как второй выход сети δ_t даст рекомендованную долю капитала в игре на данном шаге.

Поскольку, однако, в соответствии с предыдущим анализом, эта доля должна быть пропорциональна степени уверенности предсказания, можно заменить два выхода сети – одним, положив $\delta_t = \delta |y_t|$, и ограничиться оптимизацией всего одного глобального параметра δ, минимизирующего ошибку:

$$E = -\langle \ln[1 + \delta x_t y_t] \rangle .$$

Тем самым, появляется возможность регулировать ставку в соответствии с уровнем риска, предсказываемым сетью. Игра с переменными ставками приносит большую прибыль, чем игра с фиксированными ставками. Действительно, если зафиксировать

ставку, определив ее по средней предсказуемости, то скорость роста капитала будет пропорциональна $\langle \varepsilon \rangle^2$, тогда как если определять оптимальную ставку на каждом шаге, то – пропорциональна $\langle \varepsilon \rangle^2 \geq \langle \varepsilon \rangle^2 - 1$.

Приведенные выше примеры показывают, как важно уметь правильно оценить качество предсказания и как можно использовать эту оценку для увеличения прибыльности от одних и тех же предсказаний.

На следующем этапе можно пойти еще дальше и вместо среднего использовать взвешенное мнение нескольких нейронных сетей одновременно. При этом веса следует выбирать адаптивно, максимизируя предсказательную способность группы на обучающей выборке. В итоге хуже обученные сети из группы (комитета нейронных сетей) вносят меньший вклад и не портят предсказания).

Приведенные подходы планируется программно реализовать при создании универсальной системы поддержки принятия решений операторов сложных технических объектов критических областей деятельности. Апробация системы будет проводиться на примере системы оперативного обнаружения внутрисменных простоев добывающего фонда скважин нефтегазодобывающих предприятий Западной Сибири.

Литература

1. Ежов А.А., Шумский С.А. Нейрокомпьютинг и его применения в экономике и бизнесе. – Режим доступа: www.intuit.ru.
2. Информация РИА «Росбизнесконсалтинг». – Режим доступа: www.rbc.ru.
3. Гончаров М., Daniel D. Добыча знаний из CRM систем // Открытые системы. – № 3. – 2008.

Я.С. Коровін, С.Н. Матвєєв

Методика пояснення нейромережного висновку. Про підхід до розв'язання проблеми дефіциту навчальних прикладів

Запропонована і досліджена методика пояснення нейромережних розв'язків. Розглянутий підхід до розв'язання проблеми дефіциту навчальних прикладів за зразком завбачення зростання біржевих котирувань.

Ya.S. Korovin, S.N. Matveyev

A methodic of neuronetwork decision explanations' is depicted and analyzed. One solution approach of training patterns lack problem on the example of prediction of stock quotation growth is considered.

Статья поступила в редакцию 17.07.2008.